Attorney Docket No. 225682
MS # 306616.01

PATENT APPLICATION

Invention Title: Use of Static Diffie-Hellman Key with IPSEC for Authentication

Inventors:

INVENTOR'S NAME	CITIZENSHIP	CITY OF RESIDENCE	STATE or FOREIGN COUNTRY
Trevor W. Freeman	UK	Sammamish	Washington
Scott Manchester	US	Redmond	Washington
Paul G. Mayfield	US	Sammamish	Washington
Brian D. Swander	US	Bellevue	Washington

Be it known that the inventors listed above have invented a certain new and useful invention with the title shown above of which the following is a specification.

USE OF STATIC DIFFIE-HELLMAN KEY WITH IPSEC FOR AUTHENTICATION

CROSS-REFERENCE TO RELATED APPLICATION

[0001] The present application claims the benefit of Abraham et al., U.S. Provisional Patent Application No. 60/534,795 entitled, "Configuring Network Settings Using Portable Media", filed on January 7, 2004, which is hereby incorporated by reference in its entirety.

FIELD OF THE INVENTION

[0002] This invention generally relates to the area of computer systems. More particularly, the present invention concerns methods for facilitating the use of a security protocol to protect network communications, and even more particularly to methods for negotiating security parameters and authenticating users interconnected to a network.

BACKGROUND OF THE INVENTION

[0003] Computer networks provide an efficient way to exchange information between two or more computers. Various types of computer networks are utilized including private networks, e.g., local area networks (LANs), and public networks, e.g., the Internet. Often, the information exchanged between computers is of a sensitive or confidential nature. For example, to purchase goods or services via the network, a user is required to enter payment information such as a credit card number. Similarly, users routinely transmit sensitive and confidential business information over networks.

[0004] Information is exchanged over networks according to a protocol, such as the Internet Protocol (IP). IP was designed to allow for an open exchange of information; however, standard IP was not designed to protect information from unauthorized access. Accordingly, standard IP does not prevent an unauthorized user from receiving, viewing, and even modifying information transmitted over a network. Standard IP lacks other features such as authentication of users and network devices.

[0005] To address the lack of security provided by standard IP, the Internet Engineering Task Force (IETF) has developed a set of protocols, referred to as the Internet Protocol Security (IPSec) suite. IPSec provides protocols that conform to standard IP, but that include security features lacking in standard IP. Specific examples of IPSec protocols include an authentication header (AH) protocol and encapsulating security protocol (ESP). The ESP protocol, documented mainly in IETF Request for Comments (RFC) 2406, is an authenticating and encrypting protocol that uses cryptographic mechanisms to provide integrity, source authentication, and confidentiality of data. The AH protocol, documented mainly in IETF RFC 2402, is an authentication protocol that uses a hash signature in the packet header to validate the integrity of the packet data and authenticity of the sender. RFCs 2406 and 2402 are hereby incorporated by reference in their entirety for all that they teach without exclusion of any parts thereof.

[0006] Prior to using the ESP, AH or similar protocols, a first computer and a second computer in communication over the network must negotiate a set of security parameters. The first computer begins the negotiation and is usually referred to as an initiator. The second computer is referred to as a responder because it is responding to a request from the initiator. The negotiated security parameters are stored in the initiator and the responder as one or more data structures referred to as a security association (SA). Parameters stored in the SA identify a security protocol (e.g. ESP or AH), a cryptographic algorithm used to secure communication (e.g. DES, 3DES), keys used with the cryptographic algorithm, a lifetime during which the keys are valid and the like.

[0007] One method of negotiating security parameters is by using a separate negotiation protocol. An example of a negotiation protocol is the internet key management and exchange protocol (IKE), also provided as part of IPSec and documented in IETF RFC 2409, hereby incorporated by reference in its entirety for all that it teaches without exclusion of any parts thereof. IKE is generally used to negotiate and provide authenticated cryptographic keys to be used in establishing a security association (SA) in a protected manner. As practiced today, IKE typically requires multiple messages and keys between an initiator and a responder. A first set of ephemeral Diffie-Hellman (DH) keys are exchanged to establish a confidential channel. Ephemeral keys are used a limited number of times or for a limited period of time before being discarded. A second set of information is then exchanged over the confidential channel to authenticate the parties and establish a symmetric cryptographic key. The ephemeral DH keys exchanged in existing methods are not used directly for authentication. The authentication in existing IKE implementations is mutual, in that each party authenticates the identity of the other.

[0008] The IPSec protocol is also sometimes used in Virtual Private Networks (VPNs). A VPN is a private, secured network that runs over a public, unsecured network (typically the Internet). A user connecting to a VPN typically uses a password that is used to gain access to the VPN. In some existing systems, the password is also used to compute a symmetric cryptographic key for encrypting subsequent communications between the user and the VPN. In other existing VPN systems, a group of users share a pre-determined symmetric key and password to allow authentication in IKE.

BRIEF SUMMARY OF THE INVENTION

[0009] Embodiments of the invention authenticate devices and establish secure connections between devices using static Diffie-Hellman key pairs. A first device obtains in a trusted manner a static DH public key of a second device prior to negotiation. The second device negotiates a secure connection to the first device using the static DH public key, which serves as both a claim on the second device's identity and an encryption key. Because the DH key is static, rather than ephemeral, the static DH public key can be used to establish subsequent secure, authenticated communications sessions. Furthermore, using the public DH directly for authentication allows one-way authentication, where only one party authenticates the identity of the other.

[0010] In one embodiment of the invention, a method is provided for establishing a secure communications channel and authenticating a party, for use by an initiator in an Internet Security Protocol (IPSec) negotiation, comprising initiating an Internet Key Exchange (IKE) negotiation with a responder, transmitting to the responder a public Diffie-Hellman (DH) key of the initiator, receiving from the responder a public DH key of the responder, transmitting to the responder a payload encrypted with a shared secret created from the public DH key of the responder and the private DH key corresponding to the public DH key of the initiator transmitted to the responder, receiving from the responder a payload encrypted with the shared secret, and decrypting the payload, wherein the public DH key of the responder is a claim on the identity of the responder and the shared secret is used to authenticate the identity of the responder, or the public DH key of the initiator is a claim on the identity of the initiator and the shared secret is used to authenticate the identity of the initiator. In a further embodiment, the secure communications channel is a channel in a virtual private network.

[0011] In another embodiment, a method is provided for establishing a secure communications channel and authenticating a party, for use by a responder in an Internet Security Protocol (IPSec) negotiation, comprising receiving an Internet Key Exchange (IKE) negotiation request from an initiator, transmitting to the initiator a public Diffie-Hellman (DH) key of the responder, receiving from the initiator a public DH key of the initiator, transmitting to the initiator a payload encrypted with a shared secret created from the public DH key of the initiator and the private DH key corresponding to the public DH key of the responder transmitted to the initiator, receiving from the initiator a payload encrypted with the shared secret; and decrypting the payload, wherein the public DH key of the responder is a claim on the identity of the responder and the shared secret is used to authenticate the identity of the responder, or the public DH key of the initiator is a claim on the identity of the initiator and the shared secret is used to authenticate the identity of the initiator. In a further embodiment, the secure communications channel is a channel in a virtual private network.

[0012] In another embodiment, a method is provided for establishing, between an initiator and a responder, a secure communications channel following the Internet Security Protocol (IPSec), comprising using the Internet Key Exchange (IKE) protocol, wherein a static Diffie-Hellman key-pair is used by at least one of the initiator or the responder to establish confidentiality and authentication. In one embodiment, the private DH key of the DH key-pair is used to create a claim of identity for the initiator or the responder. In a further embodiment, the secure communications channel is a channel in a virtual private network.

[0013] In still another embodiment, a system is provided for establishing a secure communications channel between networked devices comprising a first networked device generating a Diffie-Hellman (DH) key pair, a portable media device storing the DH key pair generated by the first networked device, a second networked device reading the DH key pair from the portable media device, and the second networked device using the DH key pair to ensure confidentiality and authenticity in securing a communications channel with another networked device, following the Internet Key Exchange (IKE) and Internet Security (IPSec) protocols. In a further embodiment, the secure communications channel is a channel in a virtual private network.

[0014] In yet another embodiment, a computer-readable medium including computer-executable instructions is provided for facilitating establishing a secure communications channel and authenticating a party, for execution by an initiator in an Internet Security Protocol (IPSec) negotiation, said computer-executable instructions executing the steps of initiating an Internet Key Exchange (IKE) negotiation with a responder, transmitting to the responder a public Diffie-Hellman (DH) key of the initiator, receiving from the responder a public DH key of the responder, transmitting to the responder a payload encrypted with a shared secret created from the public DH key of the responder and the private DH key corresponding to the public DH key of the initiator transmitted to the responder, receiving from the responder a payload encrypted with the shared secret, and decrypting the payload, wherein the public DH key of the responder is a claim on the identity of the responder and the shared secret is used to authenticate the identity of the responder, or the public DH key of the initiator is a claim on the identity of the initiator and the shared secret is used to authenticate the identity of the initiator. In a further embodiment, the secure communications channel is a channel in a virtual private network.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] While the appended claims set forth the features of the present invention with particularity, the invention and its advantages are best understood from the following detailed description taken in conjunction with the accompanying drawings, of which:

[0016] Figure 1 is a simplified schematic illustrating an exemplary architecture of a network device for carrying out a method in accordance with an embodiment of the present invention;

[0017] Figure 2 is an exemplary network environment including multiple network devices coupled to a network, as used in accordance with embodiments of the invention;

[0018] Figure 3 is a simplified diagram of a packet payload format used to exchange payload data, as used in accordance with embodiments of the invention;

[0019] Figure 4 is a diagram illustrating a method of two devices each using a single trusted DH key pair to authenticate the other device and establish a confidential channel, in accordance with an embodiment of the invention;

[0020] Figure 5 is a flow diagram illustrating a method used by an initiator to authenticate and establish a secure communications channel with a responder using a single DH key pair, in accordance with an embodiment of the invention; and

[0021] Figure 6 is a flow diagram illustrating a method used by a responder to authenticate and establish a secure communications channel with an initiator using a single DH key pair, in accordance with an embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

[0022] The methods and systems supporting the use of a static Diffie-Hellman key pair to authenticate devices during an IPSec protocol will now be described with respect to a number of embodiments; however, the methods and systems of the invention are not limited to the illustrated embodiments. Moreover, the skilled artisan will readily appreciate that the methods and systems described herein are merely exemplary and that variations can be made without departing from the spirit and scope of the invention.

[0023] The invention will be more completely understood through the following detailed description, which should be read in conjunction with the attached drawings. In this description, like numbers refer to similar elements within various embodiments of the present invention.The invention is illustrated as being implemented in a suitable computing environment. Although not required, the invention will be described in the general context of computer-executable instructions, such as procedures, being executed by a personal computer. Generally, procedures include program modules, routines, functions, programs, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multi-processor systems, microprocessor based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices. The term computer system may be used to refer to a system of computers such as may be found in a distributed computing environment.

[0025] Figure 1 illustrates an example of a suitable computing system environment 100 on which the invention may be implemented. The computing system environment 100 is only one example of a suitable computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the invention. Neither should the computing environment 100 be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary operating environment 100. Although one embodiment of the invention does include each component illustrated in the exemplary operating environment 100, another more typical embodiment of the invention excludes non-essential components, for example, input/output devices other than those required for network communications.

[0026] With reference to Figure 1, an exemplary system for implementing the invention includes a general purpose computing device in the form of a computer 110. Components of the computer 110 may include, but are not limited to, a processing unit 120, a system memory

130, and a system bus 121 that couples various system components including the system memory to the processing unit 120. The system bus 121 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. By way of example, and not limitation, such architectures include Industry Standard Architecture (ISA) bus, Micro Channel Architecture (MCA) bus, Enhanced ISA (EISA) bus, Video Electronics Standards Association (VESA) local bus, and Peripheral Component Interconnect (PCI) bus also known as Mezzanine bus.

[0027] The computer 110 typically includes a variety of computer readable media. Computer readable media can be any available media that can be accessed by the computer 110 and includes both volatile and nonvolatile media, and removable and non-removable media. By way of example, and not limitation, computer readable media may comprise computer storage media and communication media. Computer storage media includes volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storage of information such as computer readable instructions, data structures, program modules or other data. Computer storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (DVD) or other optical disk storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by the computer 110. Communication media typically embodies computer readable instructions, data structures, program modules or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media. The term "modulated data signal" means a signal that has one or more of its characteristics set or changed in such a manner as to encode information in the signal. By way of example, and not limitation, communication media includes wired media such as a wired network or direct-wired connection, and wireless media such as acoustic, RF, infrared and other wireless media. Combinations of the any of the above are also included within the scope of computer readable media.

[0028] The system memory 130 includes computer storage media in the form of volatile and/or nonvolatile memory such as read only memory (ROM) 131 and random access memory (RAM) 132. A basic input/output system 133 (BIOS), containing the basic routines that help to transfer information between elements within computer 110, such as during start-up, is typically stored in ROM 131. RAM 132 typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 120. By way of example, and not limitation, Figure 1 illustrates operating system 134, application programs 135, other program modules 136 and program data 137.

[0029] The computer 110 may also include other removable/non-removable, volatile/nonvolatile computer storage media. By way of example only, Figure 1 illustrates a hard disk drive 141 that reads from or writes to non-removable, nonvolatile magnetic media, a magnetic disk drive 151 that reads from or writes to a removable, nonvolatile magnetic disk 152, and an optical disk drive 155 that reads from or writes to a removable, nonvolatile optical disk 156 such as a CD ROM or other optical media. Other removable/non-removable, volatile/nonvolatile computer storage media that can be used in the exemplary operating environment include, but are not limited to, magnetic tape cassettes, flash memory cards, digital versatile disks, digital video tape, solid state RAM, solid state ROM, SmartCards, SecureDigital cards, SmartMedia cards, CompactFlash cards and the like. The hard disk drive 141 is typically connected to the system bus 121 through a non-removable memory interface such as interface 140, and magnetic disk drive 151 and optical disk drive 155 are typically connected to the system bus 121 by a removable memory interface, such as interface 150.

[0030] The drives and their associated computer storage media, discussed above and illustrated in Figure 1, provide storage of computer readable instructions, data structures, program modules and other data for the computer 110. In Figure 1, for example, hard disk drive 141 is illustrated as storing operating system 144, application programs 145, other program modules 146 and program data 147. Note that these components can either be the same as or different from operating system 134, application programs 135, other program modules 136, and program data 137. Operating system 144, application programs 145, other program modules 146, and program data 147 are given different numbers hereto illustrate that, at a minimum, they are different copies. A user may enter commands and information into the computer 110 through input devices such as a tablet, or electronic digitizer, 164, a microphone 163, a keyboard 162 and pointing device 161, commonly referred to as a mouse, trackball or touch pad. Other input devices (not shown) may include a joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 120 through a user input interface 160 that is coupled to the system bus, but may be connected by other interface and bus structures, such as a parallel port, game port or a universal serial bus (USB). A monitor 191 or other type of display device is also connected to the system bus 121 via an interface, such as a video interface 190. The monitor 191 may also be integrated with a touch-screen panel or the like. Note that the monitor and/or touch screen panel can be physically coupled to a housing in which the computing device 110 is incorporated, such as in a tablet-type personal computer. In addition, computers such as the computing device 110 may also include other peripheral output devices such as speakers 197 and printer 196, which may be connected through an output peripheral interface 194 or the like.

[0031] The computer 110 is operatble in a networked environment using logical connections to one or more remote computers, such as a remote computer 180. The remote computer 180 may be a personal computer, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the computer 110, although only a memory storage device 181 has been illustrated in Figure 1. The logical connections depicted in Figure 1 include a local area network (LAN) 171 and a wide area network (WAN) 173, but may also include other networks. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet. For example, in the present invention, the computer 110 may comprise the source machine from which data is being migrated, and the remote computer 180 may comprise the destination machine. Note however that source and destination machines need not be connected by a network or any other means, but instead, data may be migrated via any media capable of being written by the source platform and read by the destination platform or platforms.

[0032] When used in a LAN networking environment, the computer 110 is connected to the LAN 171 through a network interface or adapter 170. Alternatively, the computer 110 contains a wireless LAN network interface operating on, for example, the 802.11b protocol, allowing the computer 110 to connect to the LAN 171 without a physical connection. When used in a WAN networking environment, the computer 110 typically includes a modem 172 or other means for establishing communications over the WAN 173, such as the Internet. The modem

172, which may be internal or external, may be connected to the system bus 121 via the user input interface 160 or other appropriate mechanism. Alternatively, the computer 110 contains a wireless WAN network interface operating over, for example, the General Packet Radio Service (GPRS), allowing the computer 110 to connect to the WAN 173 without a physical connection. In a networked environment, program modules depicted relative to the computer 110, or portions thereof, may be stored in the remote memory storage device. By way of example, and not limitation, Figure 1 illustrates remote application programs 185 as residing on memory device 181. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be used. Additionally, variations of the computer 110 may be incorporated into other exemplary systems for implementing the invention, such as cellular phones, personal digital assistants, and the like.

[0033] Figure 2 illustrates an exemplary network environment wherein the present invention is employed. Embodiments of the invention are directed to a method for authenticating and establishing a secure communications channel between devices through one or more networks using a static DH key pair. Embodiments are implemented as extensions to existing protocols, such as the Internet key exchange and management protocol (IKE). Alternatively, embodiments are implemented as separate proprietary protocols.

[0034] The environment includes a plurality of network devices 202, 204, 206 communicatively coupled to a network 208. The network 208 is any suitable type such as a local area network (LAN), wide area networks (WAN), intranet, the Internet, or any combination thereof. For the purpose of illustrating the invention, only a limited number of network devices are shown. However, it will be understood that many network devices may, in fact, be coupled to the network. Moreover, although the network devices are illustrated as coupled directly to the network 208, the network devices are alternatively coupled to the network 208 through a combination of servers, routers, proxies, gateways, network address translation devices, or the like.

[0035] The network device 202 communicates, i.e., exchanges information, with the network device 204 by sending packets of data according to a protocol such as the Internet Protocol (IP). The network device 202, referred to herein as the initiator, begins the exchange of information by sending a request to the network device 204, referred to herein as the responder. The network device 206 is a malicious user that attempts to gain unauthorized access to the information exchanged between the initiator 202 and the responder 204. The malicious user 206 also attempts to mount attacks on one or more of the initiator 202 and the responder 204 through, for example, a denial of service attack.

[0036] The initiator 202 includes a security policy 216 stored in security policy database. The security policy 216 is used by the initiator 202 to determine whether data transmitted to, or received from, another network device, such as the responder 204, needs to conform to a security protocol such as the Encapsulating Security Protocol (ESP) or Authentication Header (AH). The responder 204 includes its own security policy 220 stored in a security policy database that is used by the responder 204 to determine whether data transmitted to, or received from, another device, such as the initiator 202, needs to conform to a security protocol.

[0037] Security protocols such as AH and ESP protect the contents of data in an IP packet from the attacker 206. Before the security protocol is used to exchange data, the initiator 202 and the responder 204 must negotiate security parameters. The negotiated security parameters include an identification of the security protocol to be used (e.g. AH or ESP), an encryption algorithm that will be used to secure the data (e.g. DES or 3DES), keys used with the encryption algorithm to protect the data, life time that the keys will be valid and the like. The negotiated security parameters are stored in one or more data structures called a Security Association (SA).

[0038] Embodiments of the invention provide a method to exchange keys to authenticate identity and establish a confidential channel. The method is executed via a main mode 210 or an aggressive mode 212 by negotiation module 218 executing in the initiator 202 and negotiation module 222 executing in the responder 204. Each mode includes one or more pair of exchanges between the initiator 202 and the responder 204. Each exchange includes a first message sent from the initiator 202 to the responder 204 and a second message sent from the responder 204 to the initiator 202.

[0039] A main mode 210 or aggressive mode 212 is used to perform machine authentication, negotiate security parameters, and to provide a secure channel for subsequent communication under the IPSec protocol. Unlike existing systems, embodiments of the invention allow both one-way and mutual authentication. One-way machine authentication is used to prove the identity of one of, but not both, the initiator 202 and the responder 204. Mutual machine authentication is used to prove the identity of both the initiator 202 and the responder 204.

[0040] Keys are derived and refreshed with IPSec protocols such as ESP and AH. Typically, the keys used with AH and ESP to encrypt and decrypt data have a limited lifetime defined by the SA, referred to as life time of the key. Thus, it is necessary for the initiator 202 and responder 204 to periodically refresh keys used as part of the security protocol.

[0041] Figure 3 illustrates an example of a packet 228, referred to herein as a message, used to exchange data between the initiator 202 and the responder 204. The packet or message 228 includes a header portion 230 and one or more payloads 232. The format illustrated in Figure 3 generally conforms to the IKE protocol. It will be understood that the format described is by way of example, and not limitation, as any suitable format can be used to exchange data between the initiator 202 and the responder 204.

[0042] The header 230 includes an Initiator Cookie (I-Cookie) 236 and a Responder Cookie (R-Cookie) 238. The I-Cookie is a non-zero value assigned by the initiator 202 and the R-Cookie is a non-zero value assigned by the responder 204. It will be understood that the header is shown in simplified form and may include fields for additional data such as version data, flags, and a message length.

[0043] Each of the one or more payloads 232 includes a payload length field and a corresponding payload data field. The payload length field stores the size, e.g. in bytes, of the corresponding payload data. The payload data field stores data that varies depending on a payload type. The payload types included in the message depend upon the mode (e.g. main mode 210 or aggressive mode 212), state of the negotiation process, and security options employed by the initiator 202 and the responder 204. The different payload types and corresponding payload data are described in Table 1, below.

Payload Type	Payload Data
HDR	An ISAKMP header.
HDR*	An encrypted ISAKMP header.
security association (SA)	The security association includes either proposed or agreed upon security parameters.
key exchange data (KE)	Data for a key exchange according to known methods such as a Diffie-Hellman key exchange or elliptical curve.
Main mode nonce (N)	Pseudo random number sent for signing during a main mode exchange.
Kerberos authentication data (SSPI)	Kerberos authentication data also referred to as GSSAPI.
Authentication data (AUTH)	A calculated value that incorporates a secret key.
Certificate (CERT)	Includes data that establishes a user's credentials to another user such as a name, serial number, expiration date, and public key.
Certificate Request (CERTreq)	Request for a network device to provide a certificate.
Identity payload (Id)	Data that identifies a network device, such as an IP address, domain (DNS) name, or fully-qualified domain name (FQDN).
Traffic selector (TS)	Identifies transmitted or received messages subject to

Payload Type	Payload Data
	a security policy.
Vendor Id (V-Id)	Generic data field that includes data to be transmitted from a first network device to a second network device.
Notify	Generic data field that includes data to be transmitted from a first network device to a second network device.

Table 1

[0044] The payload types described in Table 1 are identified herein with the subscript "*i*" to represent values associated with the initiator 202 and with the subscript "*r*" to represent values associated with the responder 204 where appropriate. For example, N_i identifies a main mode nonce generated by the initiator 202 and N_r identifies a main mode nonce generated by the responder 204.

[0045] Returning to Figure 3, the message 228 may include a plurality of payloads and each payload has different payload data. The payload data is in the form of one of the payload types previously described herein. As shown, a first payload has a payload length 240 and corresponding first payload data 242; a second payload has a payload length 244 and corresponding second payload data 246; and a last payload has a last payload length 248 and corresponding last payload data 249.

[0046] The payloads are shown in simplified form and it will be understood that each payload may include additional information, such as data that identifies the payload types included therein.

[0047] Figure 4 illustrates a method for conducting an authentication and security negotiation between the initiator 202 and the responder 204 according to an embodiment of the invention. As previously described, the negotiation is executed by the negotiation module 218 of the initiator 202 and the negotiation module 222 of the responder 204 in accordance with the respective security policies 216 and 220.

[0048] The method is performed in a main mode 210 or an aggressive mode 212. The main mode 210 and the aggressive mode 212 are completed through a plurality of messages exchanged between the initiator 202 and the responder 204. Messages 250, 252 and 256 are messages sent from the initiator 202 to the responder 204. Messages 251, 254 and 258 are messages sent from the responder 204 to the initiator 202.

[0049] As a precursor to performing the method, the initiator and responder obtain each other's public (but not private) Diffie-Hellman (DH) keys in a trusted manner, allowing for subsequent mutual authentication. Alternatively, only one party obtains public DH keys of the other party in a trusted manner, allowing for subsequent one-way authentication. Trust in a set of DH keys can come from another mechanism, such as an Out of Band configuration. Alternatively, trust in a set of DH keys can come via an exchange through a trusted hardware device, such as a portable media device. In this embodiment, a set of DH keys are generated by a first device and stored onto a portable media device such as a USB flash drive, Secure Digital card, SmartCard, or the like. When the portable media device is attached to a second device, the second device trusts that the DH keys originated from the first device. Trust in individual DH keys can come, for example, via local policy. Alternatively, devices may employ transitive trust, where a third party endorses trust in a public DH key. Each public DH key is generated along with a corresponding private DH key.

[0050] A typical use of DH keys includes the initiator 202 exchanging public DH keys with the responder 204. The initiator 202 creates a "shared secret" from the responder's 204 public key and its own private key. The responder 204 creates the shared secret using the initiator's 202 public key and its own private key. The mathematical properties of the Diffie-Hellman algorithm guarantee that the shared secret is identical for the initiator 202 and the responder 204. Although the shared secret is useful in establishing a secure connection, it is often computationally inefficient for continued use during a communications session. The initiator 202 therefore typically creates a symmetric key for use in a more efficient cryptographic protocol such as Triple-DES, encrypts the symmetric key using the shared secret, and sends the encrypted symmetric key to the responder 204. The responder 204 decrypts the symmetric key, which is then used for efficient encrypting and decrypting during the communications session.

[0051] The method used in the embodiment begins when the initiator 202 sends message 250 to the responder 204. The message 250 is an initiation request for a key exchange negotiation following the IKE protocol. The message 250 contains an ISAKMP header and a proposed SA negotiation payload. ISAKMP is a security association management protocol, and provides a framework for SA management. The responder 204 receives the request 250 and responds with an acknowledgement message 251 containing an ISAKMP header and an agreed-upon SA negotiation payload. The agreed-upon SA includes security parameters, selected from the proposed security parameters, to which the responder 204 agrees. If the responder does not agree to a set of the parameters in the proposed SA, the negotiation fails.

[0052] The initiator 202 sends message 252 to the responder 204. The message 252 has a plurality of payload types including an ISAKMP header, a nonce (Ni), and a key exchange payload (KE). The KE comprises a public DH key for the initiator 202. Because the responder 204 has previously obtained the initiator's 202 public DH key in a trusted manner, the KE acts as a claim for the initiator's 202 identity. The initiator 202 creates a shared secret from its private key and the responder's 204 public key.

[0053] The responder 204 receives the message 252 and in return sends the message 254 back to the initiator 202. The message 254 has a plurality of payload types including a ISAKMP header, a responder nonce (N_r) and key exchange data (KE). The KE comprises a public DH key for the responder 204. Because the initiator 202 has previously obtained the responder's 204 public DH key in a trusted manner, the KE acts as a claim for the responder's 204 identity. The responder 204 creates the shared secret from its private key and the initiator's public key. This shared secret is, by mathematics, identical to the shared secret produced by the initiator 202.

[0054] The initiator 202 receives the message 254 and, in return, sends the message 256 to the responder 204. The message 256 has a header (HDR*) and a plurality of payload types including an identification payload (IDii) and a hash payload (HASH_I). The HDR* header indicates the payload is encrypted using the shared secret.

[0055] The responder 204 receives the message 256 and decrypts the encrypted payload using the shared secret. Because only a holder of the private DH key can encrypt a message that is decrypted with the corresponding public DH key, and the responder 204 trusts that the public DH key previously sent belongs to the initiator 202, the responder 204 is convinced that it is in communication with the identity of initiator 202. The responder 204 replies with a message 258 to the initiator 202. The message 258 has a header (HDR*) and a plurality of payload types including an identification payload (IDir) and a hash payload (HASH_R). The HDR* header indicates the payload is encrypted using the shared secret.

[0056] The initiator 202 receives the message 258 and decrypts the encrypted payload using the shared secret. The initiator is thereby convinced of the identity of the responder 204.

[0057] In an alternative embodiment of the described method, one-way authentication is provided. In this embodiment, one device learns the public DH key of the second device in a trusted manner as described above, prior to beginning the method. The second device does not, however, know the public DH key of the first device. For the following example, the initiator 202 has learned the public DH key of the responder 204 in a trusted manner. The method proceeds as above, with the initiator 202 sending an initiation request message 250 to the responder 202. The responder 202 responds with a message 251.

[0058] The initiator then sends message 252 as above. However, because the responder 204 does not know the public DH key of the initiator 202, the KE payload in message 252 does not serve as an identity claim on the initiator 202. The initiator 202 thus remains anonymous to the responder 204.

[0059] The responder 204 responds to message 252 by sending message 254. The KE payload in message 254 contains the public DH key for the responder 204. Because the initiator 202 has previous knowledge of this public DH key, the key serves as an identity claim on the responder 204.

[0060] The initiator 202 receives the message 254 and, in return, sends the message 256 to the responder 204. The message 256 includes an HDR*, which indicates the payload is encrypted using the shared secret created from the responder's 204 public DH key and the initiator's private DH key.

[0061] The responder 204 receives the message 256 and decrypts the encrypted payload using the shared secret created from its private key and the initiator's 202 public key. Because the responder 204 does not know the identity of the owner of the DH keys used for this encryption, the responder is not convinced of the initiator's 202 identity. The responder 204 replies with a message 258 to the initiator 202. The message 258 includes an HDR* header indicating that the payload is encrypted using the shared secret.

[0062] The initiator 202 receives the message 258 and decrypts the encrypted payload using the shared secret. The initiator 202 is thereby convinced of the identity of the responder 204.

[0063] In an alternative example, the responder 204 has learned the public DH key of the initiator 202 in a trusted manner prior to beginning the method, and the method proceeds similarly as described above, with the responder 204 being convinced of the identity of the initiator 202..

[0064] An embodiment of the invention performs a variation of the method in an aggressive mode. In the aggressive mode, three exchanges take place. First, the initiator 202 passes to the responder 204 a message containing a ISAKMP header, a proposed SA negotiation payload, a KE comprising a public key for the initiator 202, a nonce (Ni), and an identification payload IDii. Second, the responder 204 replies with a ISAKMP header, an accepted SA negotiation payload, a KE comprising a public key for the responder 204, a nonce (Nr), and identification payload IDir, and a hash payload (HASH_R). The initiator 202 and responder 204 create a shared secret using the exchanged public keys and their own private keys. If either of the exchanged public keys had been previously obtained in a trusted manner, it serves as a claim to the owner's identity. The third exchange is when the initiator 202 replies with a header (HDR*) and a hash payload (HASH_I), encrypted using the shared secret.

[0065] Turning attention to Figure 5, a method is described for use by an initiator to authenticate and establish a secure communications channel with a responder using a static DH key pair, in accordance with an embodiment of the invention. The method begins by initiating an IKE negotiation at step 502. The initiation preferably comprises sending a ISAKMP header and proposed SA negotiation payload. In response to the negotiation request, the initiator receives a reply at step 504. The initiator checks that the SA negotiation was successful at step 506. If the negotiation was not successful and a security association was not agreed upon, then the initiator starts another negotiation initiation at step 502. Alternatively, the negotiation process terminates. Otherwise, the initiator continues by sending a key exchange (KE) payload at step 508. The KE payload comprises the initiator's public DH key. In one embodiment, the responder has previously obtained the initiator's public DH key in a trusted manner. In such an embodiment, the KE sent at step 508 acts as a claim to the initiator's identity.

[0066] The initiator receives a reply at step 509. The reply contains a public DH key for the responder. In one embodiment, the initiator has previously obtained the responder's public DH key in a trusted manner. In such an embodiment, the KE sent at step 509 acts as a claim to the responder's identity.

[0067] The initiator sends an encrypted payload at step 510. The payload is encrypted using a shared secret created from the initiator's 202 private key and the public key received at step 509.

[0068] The initiator receives a message at step 512 comprising an encrypted payload, which the initiator decrypts using the shared secret. The initiator decides whether or not to trust the identity of the responder at step 514 by noting if the public key received at step 509 corresponds to a previously known, trusted identity. If so, the initiator trusts the responder's identity at step 516. Otherwise, the initiator does not trust the identity of the responder at step 518.

[0069] Turning attention to Figure 6, a method is described for use by a responder to authenticate and establish a secure communications channel with an initiator using a static DH key pair, in accordance with an embodiment of the invention. The method begins by receiving an IKE negotiation at step 602. The initiation preferably comprises receiving a ISAKMP header and proposed SA negotiation payload. In response to the negotiation request, the responder sends a reply at step 604 preferably comprising a ISAKMP header and an accepted SA negotiation payload.

[0070] The responder receives a key exchange (KE) payload at step 608. The payload contains a public DH key for the initiator. In one embodiment, the responder has previously obtained the initiator's public DH key in a trusted manner. In such an embodiment, the KE sent at step 608 acts as a claim to the initiator's identity.

[0071] The responder continues by sending a key exchange (KE) payload at step 609. The KE payload comprises the responder's public DH key. In one embodiment, the initiator has previously obtained the responder's public DH key in a trusted manner. In such an embodiment, the KE sent at step 609 acts as a claim to the responder's identity.

[0072] The responder receives a message at step 610 comprising an encrypted payload, which the responder decrypts using a shared secret created from the responder's 204 private key and the public DH key received at step 608.

[0073] The responder sends an encrypted payload at step 612. The payload is encrypted using the shared secret.

[0074] The responder decides whether or not to trust the identity of the initiator at step 614 by noting if the public key received at step 610 corresponds to a previously known, trusted identity. If so, the responder trusts the initiator's identity at step 616. Otherwise, the responder does not trust the identity of the initiator at step 618.

[0075] The above-described methods are further applicable in a Virtual Private Network setting. Using a static DH key in the manner described above, a device connects and is authenticated to a VPN via the IKE protocol. This is used to establish a secure channel that follows the IPSec protocol. Typically, in existing VPNs, the client and server use a shared key derived from a hash of the client's user-password. Because the identity of the client is encrypted, existing VPN servers are limited by a conundrum: they need to know the identity of the client in order to use the correct shared key for decryption.

[0076] In accordance with an embodiment of the invention, a VPN server (responder) first obtains the public DH key of a client (initiator) in a trusted manner, prior to beginning a negotiation. To begin a negotiation, a client sends his public DH key to the server. The public DH key acts as a hint to the user's identity. Using this hint, the server finds the corresponding shared key for the user, and performs pre-shared key authentication with the client using the shared key.

[0077] In view of the many possible embodiments to which the principles of the present invention may be applied, it should be recognized that the embodiments described herein with respect to the drawing figures are meant to be illustrative only and should not be taken as limiting the scope of the invention. For example, those of skill in the art will recognize that the illustrated embodiments can be modified in arrangement and detail without departing from the spirit of the invention. Although the invention is described in terms of software modules or components, those skilled in the art will recognize that such may be equivalently replaced by hardware components. Devices embodying the invention can include, for example, computers, personal digital assistants (PDAs), portable media devices (USB flash drives, Smartcards, etc.), and the like. Therefore, the invention as described herein contemplates all such embodiments as may come within the scope of the following claims and equivalents thereof.